Neurocomputing 107 (2013) 3–10

Contents lists available at SciVerse ScienceDirect

Neurocomputing

journal homepage: www.elsevier.com/locate/neucom

Training spiking neural networks to associate spatio-temporal input–output spike patterns

Ammar Mohemmed [a,*], Stefan Schliebs [a], Satoshi Matsuda [c], Nikola Kasabov [a,b]

[a] Knowledge Engineering and Discovery Research Institute, Auckland University of Technology, Private Bag 92006, Auckland 1142, New Zealand
[b] Institute for Neuroinformatics, ETH and University of Zurich, Switzerland
[c] Department of Mathematical Information Engineering, Nihon University, Japan

ARTICLE INFO

Available online 23 October 2012

Keywords:
Spiking neural networks
Supervised learning
Spatio-temporal control
Temporal coding

ABSTRACT

In a previous work (Mohemmed et al., Method for training a spiking neuron to associate input–output spike trains) [1] we have proposed a supervised learning algorithm based on temporal coding to train a spiking neuron to associate input spatiotemporal spike patterns to desired output spike patterns. The algorithm is based on the conversion of spike trains into analogue signals and the application of the Widrow–Hoff learning rule. In this paper we present a mathematical formulation of the proposed learning rule. Furthermore, we extend the application of the algorithm to train a SNN consisting of multiple spiking neurons to perform spatiotemporal pattern classification and we show that the accuracy of classification is improved significantly over a single spiking neuron. We also investigate a number of possibilities to map the temporal output of the trained spiking neuron into a class label. Potential applications for motor control in neuro-rehabilitation and neuro-prosthetics are discussed as a future work.

1. Introduction

The perfection exhibited by living entities in carrying out their daily natural activities is inspiring researchers to adopt their behavior, deep to the cell level, as a model to solve computational tasks that are considered complex for machines to solve. The study of spiking neural networks (SNN) [2–5] represents a significant step in the path of learning from the brain. SNN is closer to the real operational model of the brain than conventional neural networks. This closeness is asserted in the use of spikes as a form of communication between the neural nodes similar to the brain. The shape of the spike seems less relevant and has no importance in representing the information, instead the time of spiking carries the information. How information is encoded in the spike timing is a debatable issue as many theories exist. Traditionally, the commonly used neural code in SNN is rate coding in which the information is encoded in the number of spikes over a small time window. Alternatively, the temporal coding encodes the information in the exact timing of the spikes. Information representation has an important role in simplifying and speeding the computation to achieve good results. In [6] it was argued that the recognition of patterns such as colors, visual patterns, odors and sound quality are solved rapidly in neurobiology using temporal coding and could not be solved using rate-based neural models. Furthermore, temporal coding is supported by evidences observed in different types of biological neurons, see [7] for a survey.

The other issue establishing the biological plausibility of SNN is the learning paradigm referred to as spike time dependent plasticity (STDP) [8,9,2,10]. The STDP is an unsupervised learning process that adjusts the synaptic weights based on the time correlation between the incoming spike (presynaptic spike) and the emitted spike of the neuron (postsynaptic spike). In [11] it was shown that STDP enables a neuron to perform a complex recognition task: to localize a repeating spatiotemporal spike pattern embedded in equally dense distractor spike trains. In [12], an unsupervised learning algorithm based on STDP and Winner-Take-All (WTA) paradigm is proposed for pattern recognition.

However, for specific task oriented engineering applications, supervised learning (training) or a combined unsupervised–supervised might be more favorable over unsupervised learning. Supervised learning, commonly in the form of error back propagation [13], is widely used in training conventional neural networks to perform pattern recognition. Due to the nature of spike-based communication and the complexity of SNN (which requires tuning big number of parameters), no efficient supervised learning techniques for SNN have existed until recently.

One of the first supervised learning methods for SNN is SpikeProp [14]. This uses a gradient descent approach that adjusts

* Corresponding author.
E-mail addresses: amohemme@aut.ac.nz (A. Mohemmed), sschlieb@aut.ac.nz (S. Schliebs), matsuda.satoshi@nihon-u.ac.jp (S. Matsuda), nkasabov@aut.ac.nz (N. Kasabov).
URL: http://www.kedri.info (A. Mohemmed).

0925-2312/$ - see front matter © 2012 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.neucom.2012.08.034

the synaptic weights in order to emit a single spike at a specified time. The timing of the output spike encodes specific information, e.g. the class label of the presented input sample. However, SpikeProp cannot train SNN to emit a desired spike train consisting of more than one spike.

An interesting learning rule for spatiotemporal pattern recognition has been suggested in [12]. The so-called Tempotron enables a neuron to learn whether to fire or not to fire in response to a specific input stimulus. Consequently, the method allows the processing of binary classification problems. However, the neuron is not intended to learn a precise target output spike train, but instead whether to spike or not to spike in response to an input stimulus.

A Hebbian based supervised learning algorithm called remote supervised method (ReSuMe) was proposed in [15] and further studied in [16,17]. ReSuMe, similar to STDP, is based on a learning window concept. Using a teaching signal a specific desired output is imposed on the output neuron. With this method, a neuron can produce a spike train precisely matching a desired spike train. It was shown that in combination with the liquid state machine (LSM) [18], the algorithm is efficient for random mapping from any input spike train to any output spike train or multiple spike trains. The algorithm was mainly designed and applied for neuroprostheses control [19].

Recently, a method called Chronotron was proposed [20]. Two versions of learning rules are described therein; E-learning and I-learning. E-learning is based on minimizing the error between the desired spike pattern and the actual one. The error is measured using the Victor–Purpura spike distance metric [21]. This metric produces discontinuities in the error landscape that must be overcome through approximation. E-Learning surpasses ReSuMe in terms of the number of spike patterns that can be memorized and classified. The other version, I-Learning, is biologically more plausible but less efficient.

In [22] the authors proposed a supervised learning paradigm for SNN based on particle swarm optimization (PSO). PSO optimizes, according to a fitness function, the parameters of the dynamic synapses [23] which connect the layers of the network. The fitness function measures the similarity between the actual output spike train and the target spike train. However, PSO becomes less efficient at finding good solutions when the number of variables (i.e. the parameters of the synapses) increases, limiting its applicability for large networks, especially when the input stimulus is a spatiotemporal spike pattern consisting of many spike trains. To overcome this difficulty, the authors proposed in [1] a simple method to train a neuron to map (associate) an input spatiotemporal spike pattern to a desired spike train pattern. The method is based on the Widrow–Hoff (or Delta) learning rule [24] commonly used in traditional neural networks. The Delta rule adjusts the weight of a synapse by scaling the error signal, i.e. the difference between the teacher signal and the actual signal, by the value of the input at that synapse. The Delta rule is inapplicable to SNN because spikes, unlike real-values signals, cannot be subtracted or multiplied directly. In the mentioned proposed learning rule, spike trains are converted into continuous signals by convolution with a kernel function. The Delta rule can then be applied directly to adjust the synaptic weight for training purposes. We refer to a spiking neuron trained via this method by SPAN (Spike Pattern Association Neuron) since the neuron is intended primarily for input/output spike pattern association. SPAN was evaluated to be efficient in a synthetic spatiotemporal classification problem [1].

In this paper, a mathematical formulation of SPAN learning rule is provided. Furthermore, instead of a single SPAN to perform spatiotemporal classification, we train multiple SPANs in a single layer network to perform the classification task and compare the accuracy with that of a single SPAN. Because SPAN is based on temporal coding, we describe and test different ways to transform the output spike pattern into a class label.

In the next section the learning rule is described and derived mathematically. In Section 3 we discuss the multiple SPAN architecture. In Section 4, the details of the simulation experiments and results using multiple SPANs are given. Section 5 concludes the paper and highlights future research and applications.

2. The SPAN learning rule

Similar to other supervised training algorithms, the synaptic weights of the network are adjusted iteratively to impose a desired input/output spike pattern association to the SNN. To derive the learning rule, we begin with Widrow–Hoff rule as follows. For a synapse i, the weight change Δw_i is defined as

$$\Delta w_i = \lambda x_i \, (y_d - y_{out}) = \lambda x_i \Delta_i \tag{1}$$

where $\lambda \in \mathbb{R}$ is a real-valued positive learning rate, x_i is the input transferred through synapse i, and y_d and y_{out} refer to the desired and the actual neural output, respectively. Note that $\Delta_i = y_d - y_{out}$ is the difference or error between the desired and the actual output of the neuron.

This rule was introduced for conventional neural networks where the input and output are real-valued signals. In SNN however, trains of spikes are passed between neurons rendering the Widrow–Hoff rule incompatible for SNN. More specifically, if x_i, y_d and y_{out} are considered as spike trains $s(t)$ defined by

$$s(t) = \sum_f \delta(t - t^f) \tag{2}$$

where t^f is the firing time of a spike and $\delta(\cdot)$ is the Dirac delta function $\delta(x) = 1$ if $x = 0$ and 0 otherwise, then the difference between two spike trains y_d and y_{out} does not define a suitable error landscape which can be minimized by a gradient descent method.

Here, we address this issue by proposing the following idea. In order to define the difference between spike trains, we convolve each spike sequence with a kernel function $\kappa(t)$. This is similar to the binless distance metric used to compare spike trains [25]. We define

$$\tilde{x}_i(t) = \sum_{t_i^f \in F_{in}} \kappa(t - t_i^f) \tag{3}$$

$$\tilde{y}_d(t) = \sum_{t_d^g \in F_d} \kappa(t - t_d^g) \tag{4}$$

$$\tilde{y}_{out}(t) = \sum_{t_{out}^h \in F_{out}} \kappa(t - t_{out}^h) \tag{5}$$

with F_{in}, F_d and F_{out} being the input, the desired and the actual output set of spike trains, respectively. Substituting x_i, y_d and y_{out} with the kernelized spike trains $\tilde{x}_i(t)$, $\tilde{y}_d(t)$ and $\tilde{y}_{out}(t)$, a new learning rule for a spiking neuron is obtained:

$$\Delta w_i(t) = \lambda \tilde{x}_i(t) \, (\tilde{y}_d(t) - \tilde{y}_{out}(t)) \tag{6}$$

This equation formulates a real-time learning rule such that the synaptic weights change over time. By integrating Eq. (6), we derive the batch version of the learning rule which is under scrutiny in this paper:

$$\Delta w_i = \lambda \int_0^\infty \tilde{x}_i(t) \, (\tilde{y}_d(t) - \tilde{y}_{out}(t)) \, \mathrm{d}t \tag{7}$$